

ASIA PACIFIC UNIVERSITY OF

TECHNOLOGY & INNOVATION

Designing & Developing Cloud Application

(CT071-3-3-DDAC)

Individual Assignment

**Name**: Nur Raihana Farhani Binti Mohd Nozeri

**TP number**: TP040647

**Due Date**: 9 October 2017

**Lecture**: DR. Kalai Anand Ratnan

# Acknowledgment

First of all, I would take the opportunity to thank my university which is Asia Pacific University (APU) to let me have this module in my final year. In this module, I can gain many new knowledge and experience about cloud computing and cloud technology like Azure. Besides, my lecture which is DR. Kalai Anand Ratnan gives me a lot useful material guide to help me in understand the requirement of assignment needed. In the tutorial class, he is showing step by step how to upload a website into Azure and how to create a database, those entire useful tutorials can save my time for exploring the way of doing the assignment.

Last but not least, I would like to thank my friend as well for helping me when I have trouble doing the assignment and give me moral support. They are giving me comment for how to enhance my assignment quality and guiding me slowly when I unable to complete the tutorial.

Table of Contents

[Acknowledgment 2](#_Toc495306089)

[1. Introduction 5](#_Toc495306090)

[1.1. Background of the project 5](#_Toc495306091)

[1.2. Objective 6](#_Toc495306092)

[1.3. Scope 6](#_Toc495306093)

[1.4. Requirement specification 6](#_Toc495306094)

[1.5. Function 7](#_Toc495306095)

[2. Project Plan 8](#_Toc495306096)

[3. Design 10](#_Toc495306097)

[3.1. Architecture Diagram 10](#_Toc495306098)

[3.2 Database Design 11](#_Toc495306099)

[3.3 Use case Diagram 12](#_Toc495306100)

[3.3.1 Use case Description 13](#_Toc495306101)

[3.4 Sequence Diagram 16](#_Toc495306102)

[3.5 Class Diagram 17](#_Toc495306103)

[4. Implementation 18](#_Toc495306105)

[4.1 Create Resource Group 18](#_Toc495306106)

[4.2 Create Web Application 21](#_Toc495306107)

[4.3 Create Traffic Manger Profile 23](#_Toc495306108)

[4.4 Upload Web Application 25](#_Toc495306109)

[4.5 Web application interface 28](#_Toc495306110)

[5. Test plan & Testing Discussion 31](#_Toc495306111)

[5.1 Unit Testing 31](#_Toc495306112)

[5.2 User Acceptance Testing 32](#_Toc495306113)

[6. Conclusion 33](#_Toc495306114)

[7. References 34](#_Toc495306115)

# Introduction

## Background of the project

Ukraine International Airlines (UIA) is the leading carrier and largest airline in Ukraine. It operates domestic and international passenger flights and cargo services to Europe, the Middle East, the United States, and Asia. Through its numerous interline agreements, UIA operates flights to over 3 000 destinations across the globe. The company is known to be founded in the year 1992.

By use sophisticated software for analysing fuel economy and used technology to reduce costs, innovate, and improve customer service long used ago, the airlines decide again to innovate by web challenges. The UIA has problems with the website that prevented it from adequately serving customers beyond Ukraine. It is severe denial-of-service (DOS) attacks and which hurt site performance and reliability, and it did not have the performance needed to host visitors from many parts of the world. To solve all the problem the website, need to migrate out of the UIA data centre which into a public cloud.

## Objective

* + - To develop an interactive management system to solve the traffic problem
    - To deploy Maersk Line system using Microsoft Azure Cloud platform
    - To develop a solution which the website prevented from adequately serving customer beyond Ukraine.

## Scope

* Able to upgrade the system
* Able to deploy in Microsoft Azure as a cloud base web application
* Able to monitor the application when user using the application
* Able to maintain the application while multiple tenants are using it
* Able to observe the application problem and create solution
* Able to rollback while the latest application has problem

## Requirement specification

* Provisioning: You must be able to provision the new application to the Microsoft Azure Platform.
* Maintainability: You must be able to upgrade the application and perform other maintenance tasks while multiple tenants are using it.
* Monitoring: You must be able to monitor the application at all times to identify any problems and to troubleshoot them. This includes monitoring how each tenant is using the application.
* Availability: Tenants want the application to be constantly available, perhaps with guarantees defined in an SLA. Again, the activities of other tenants should not affect the availability of the application.
* Scalability: The application scales to meet the demand of the application

## Function

According to UIA situation, the function of the system needed is:

**Customer Login**

To book the ticket, the customer need to login first. If the customer did not have the account, they need to register.

**Book Flight Ticket**

The system can book the flight ticket based on user wish. The destination and departure place and date will be choosing by the user.

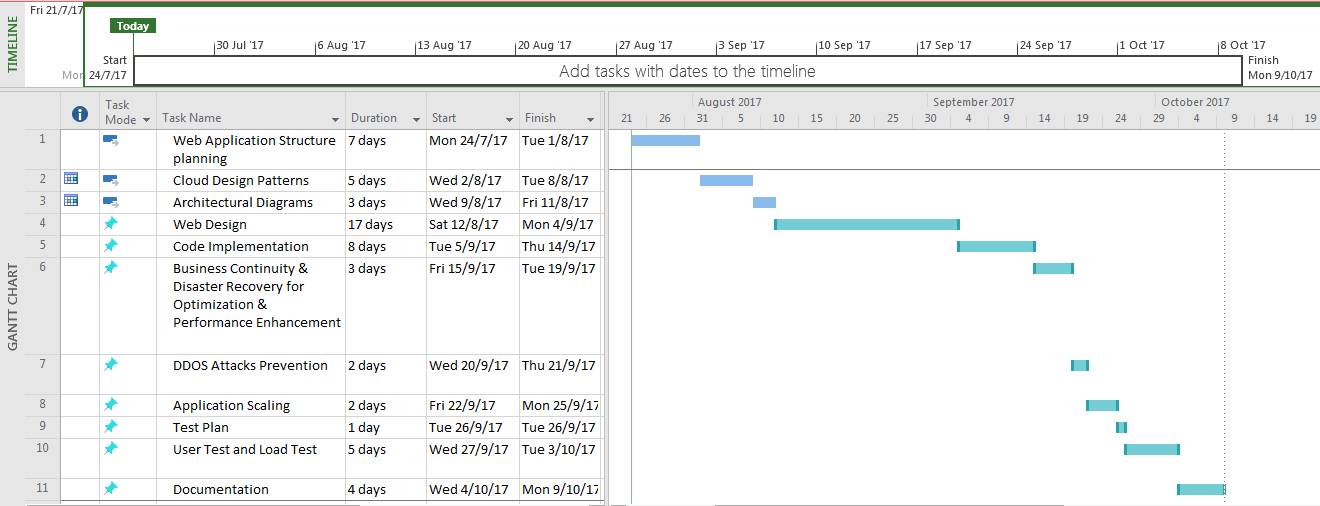
**Search Flight Ticket**

User search the flight ticket at the home page. It will be showed the flight details, information about the of the chosen destination. It will be easier for user to see the flight information.

# Project Plan

Project Plan is important for every single project because it ensure the project progress are on the track and complete the task before the deadlines. Time and budget are also a critical factor for development project. Therefore, in order to ensure that the project can delivered on time and cost efficient, a well plan project plan is needed for it. The table below is the project plan for this project, this project only works on weekday.

|  |  |
| --- | --- |
| **Task Name** | **Duration** |
| Web Application Structure planning | 7 Days |
| Cloud Design Patterns | 5 days |
| Architectural Diagrams | 3 days |
| Web Design | 17 days |
| Code Implementation | 8 days |
| Business Continuity & Disaster Recovery for Optimization & Performance Enhancement | 3 days |
| DDOS Attacks Prevention | 2 days |
| Application Scaling | 2 days |
| Test Plan | 1 days |
| User Test and Load Test | 5 days |
| Documentation | 4 days |
| **Total** | **57 days** |



# Design

## Architecture Diagram

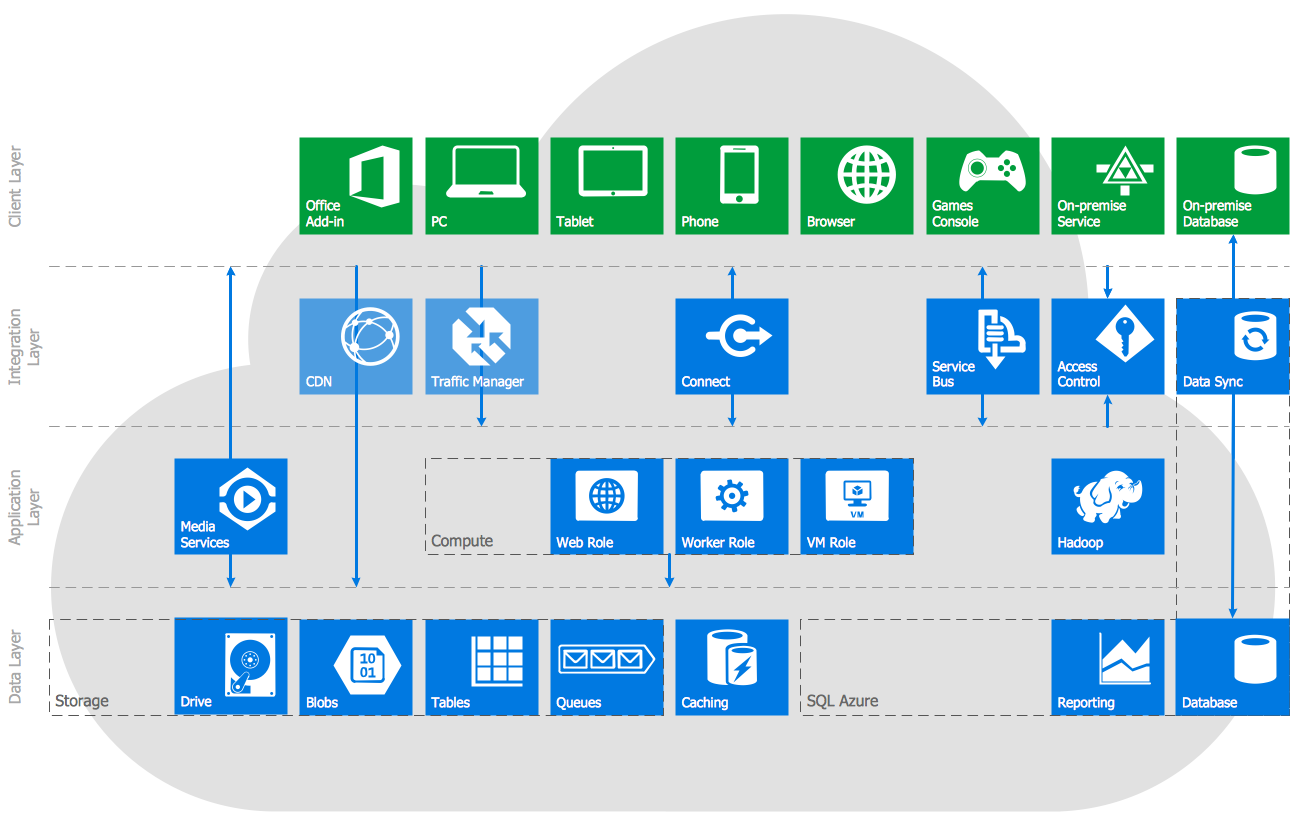


Image above is Azure Architecture diagrams solution is extending Concept Draw PRO with a comprehensive collection of Microsoft Azure themed graphics and logos, and present templates to help developer get started with the diagram

## Database Design

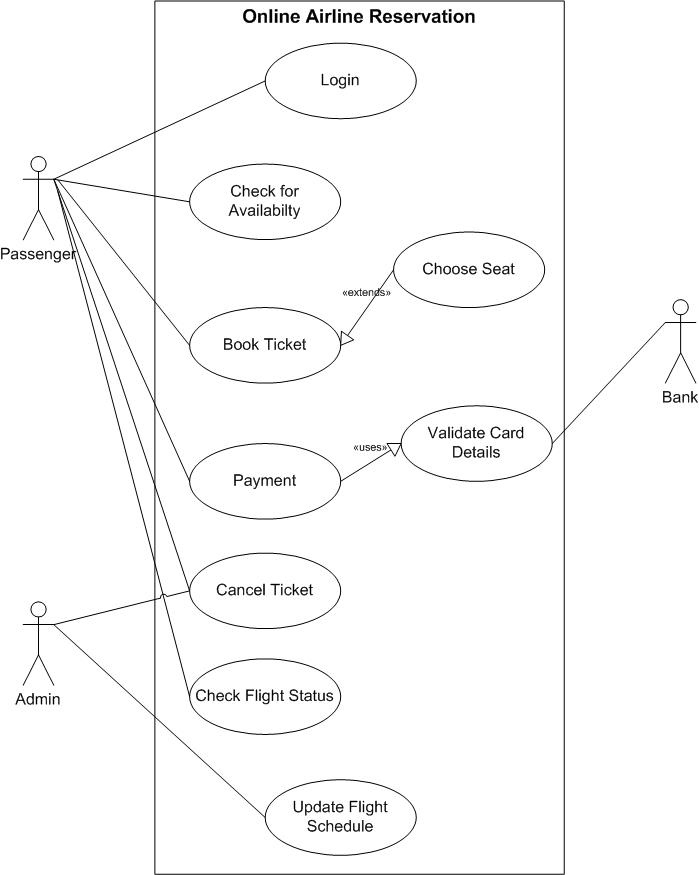
|  |  |
| --- | --- |
| Userinformation | |
| PK | UserID |
|  | Password  Name  User type |

|  |  |
| --- | --- |
| FlightSchedule | |
| PK | ScheduleID |
|  | Time  Date |

|  |  |
| --- | --- |
| FlightDetails | |
| PK | FlightID |
|  | Name  Flight type |

|  |  |
| --- | --- |
| Destination | |
| PK | DestinationID |
|  | Destination  Price |

## Use case Diagram



## Use case Description

|  |  |
| --- | --- |
| **Use Case** | Login |
| **Description** | Allow user to access to more function in the application |
| **Actor** | Passenger |
| **Pre-condition** | Register an account |
| **Functions** | This function is let passenger access the application function. The system will identify user and redirect them to home page in order to continues the functionality. |

|  |  |
| --- | --- |
| **Use Case** | Book ticket |
| **Description** | To let passenger to book flight ticket |
| **Actor** | Passenger |
| **Pre-condition** | Login |
| **Functions** | Passenger can book flight ticket |

|  |  |
| --- | --- |
| **Use Case** | Choose seat |
| **Description** | To let passenger to choose flight seat |
| **Actor** | Passenger |
| **Pre-condition** | Login |
| **Functions** | Passenger can choose seat based on their wish |

|  |  |
| --- | --- |
| **Use Case** | Payment |
| **Description** | To let passenger, make a payment |
| **Actor** | Passenger |
| **Pre-condition** | Login |
| **Functions** | Passenger can make payment to continue book flight. |

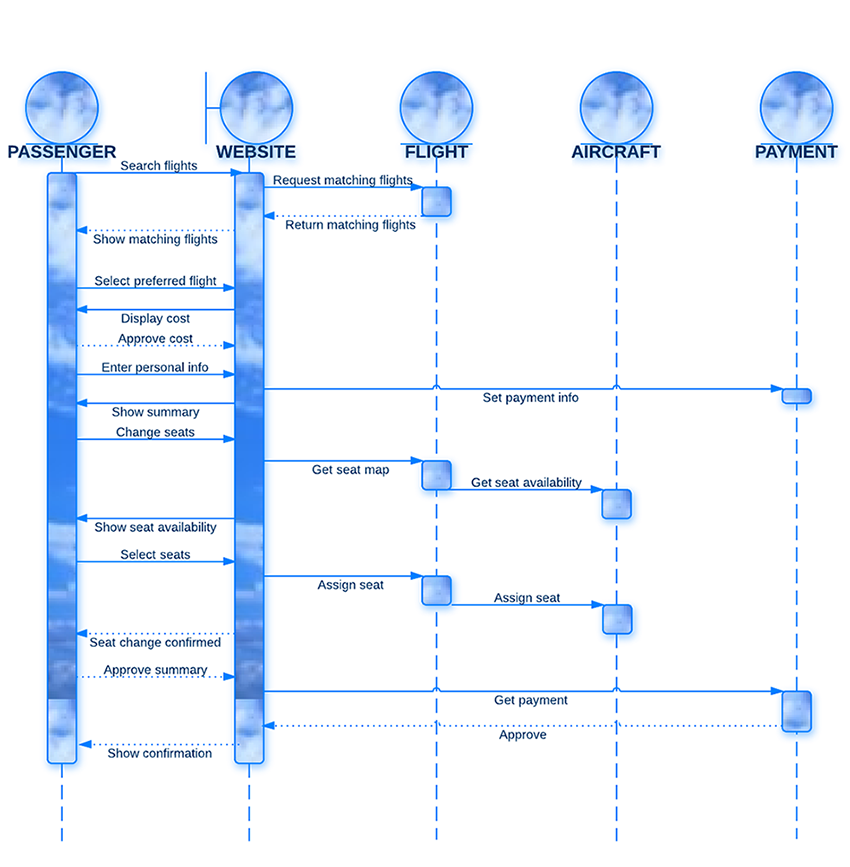
|  |  |
| --- | --- |
| **Use Case** | Cancel Ticket |
| **Description** | To let passenger and admin cancel flight ticket |
| **Actor** | Passenger, admin |
| **Pre-condition** | Login |
| **Functions** | Passenger and admin can cancel the flight ticket that has been booked. |

|  |  |
| --- | --- |
| **Use Case** | Check flight status |
| **Description** | To let passenger, check the flight status |
| **Actor** | Passenger |
| **Pre-condition** | - |
| **Functions** | Passenger can check the flight status to have more details about inbound and outbound flight. |

|  |  |
| --- | --- |
| **Use Case** | Update Flight Schedule |
| **Description** | To let admin update flight schedule |
| **Actor** | Admin |
| **Pre-condition** | Login |
| **Functions** | Admin can check the flight status to have more details about inbound and outbound flight. |

|  |  |
| --- | --- |
| **Use Case** | Validate card details |
| **Description** | To let admin update flight schedule |
| **Actor** | Bank |
| **Pre-condition** | - |
| **Functions** | Bank can check the card details of passenger to validate the payment |

## Sequence Diagram

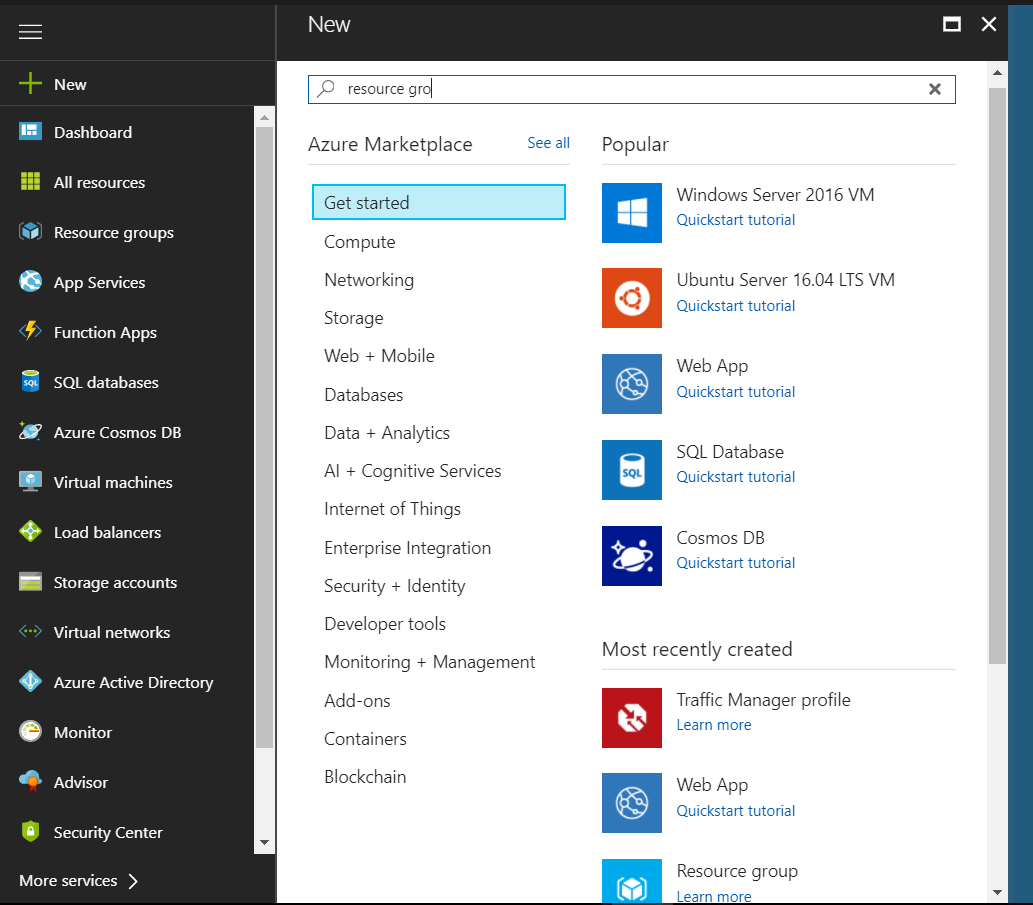


## Class Diagram

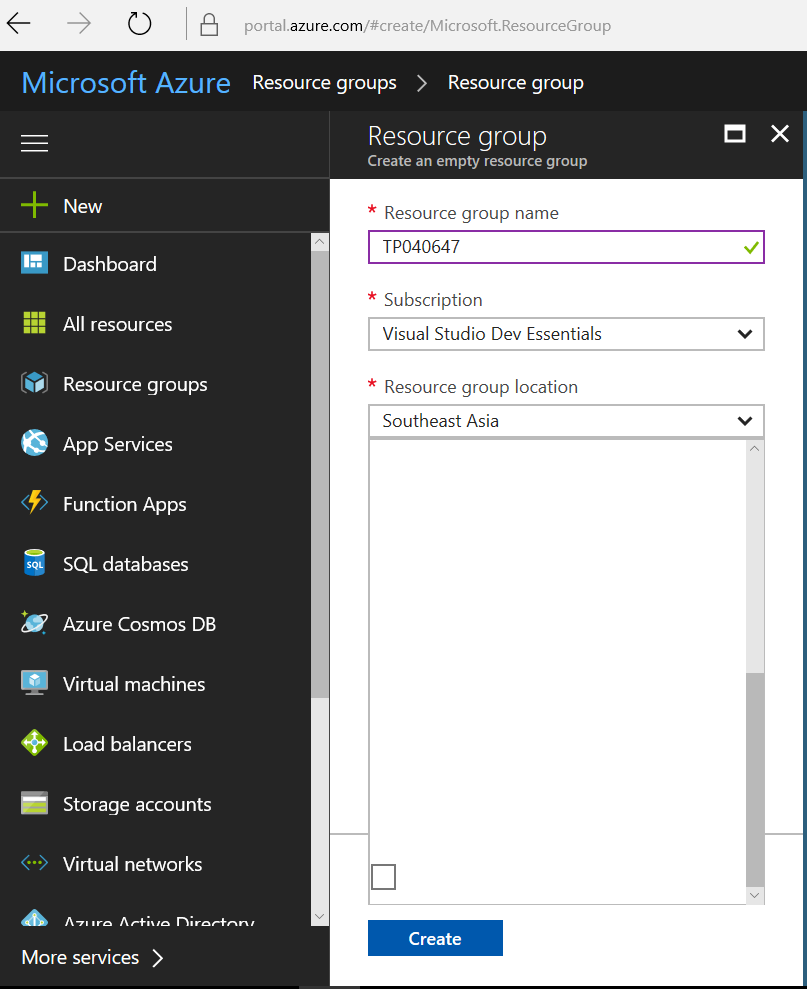
## 

# Implementation

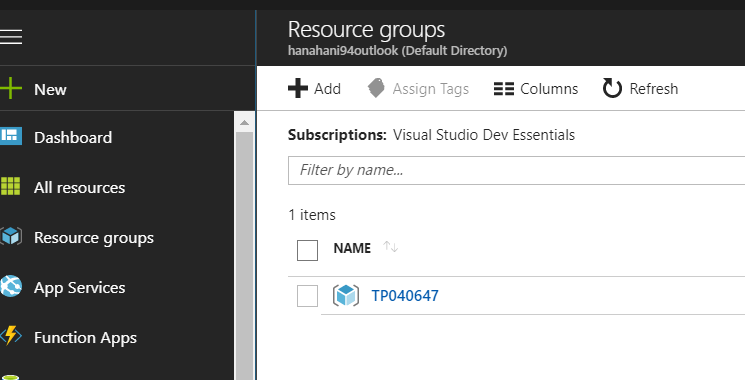
## Create Resource Group



To begin with this project must have a resource group, so developer plan to search the resource group from the new marketplace to search the resource group.

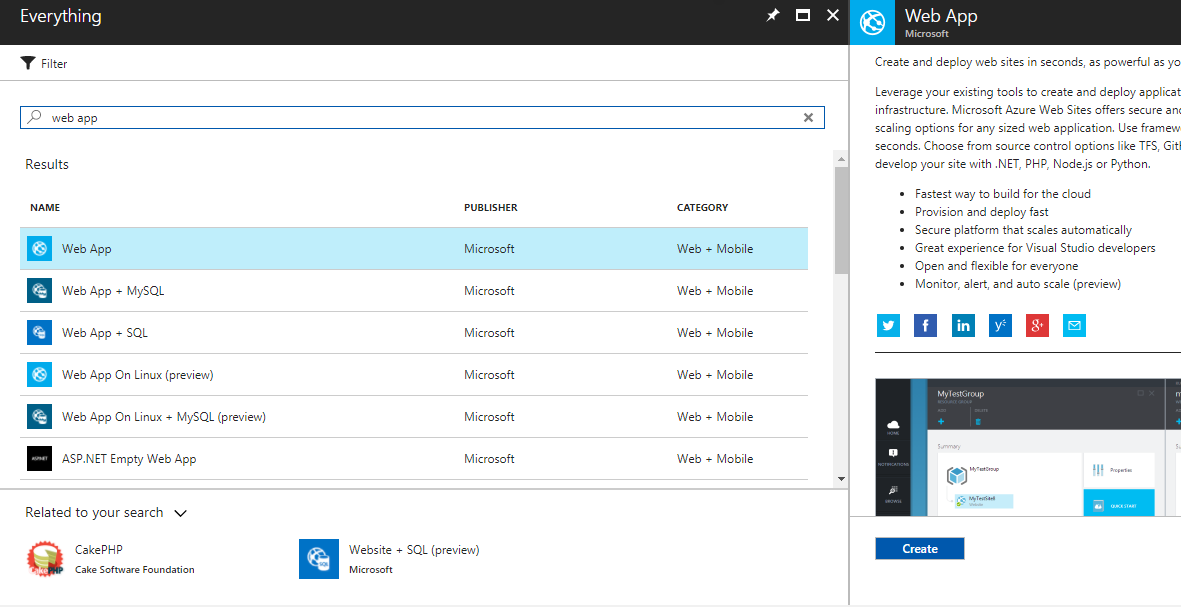


After the searching process, developer starts to create a new resource group. The new resource group require the resource group name and the resource group location. The nearest to the project location is Southeast Asia.

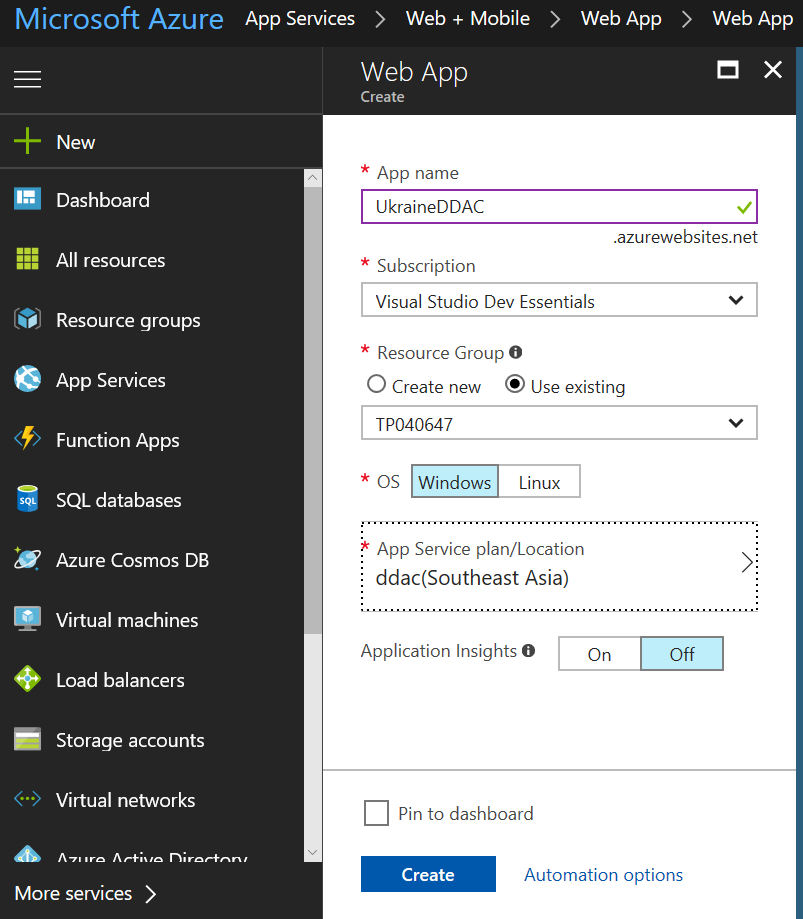


After the resource group successfully create, developer can view the resource group at resource group.

## Create Web Application



After create a resource group, than developer will go for marketplace to find web app to create.

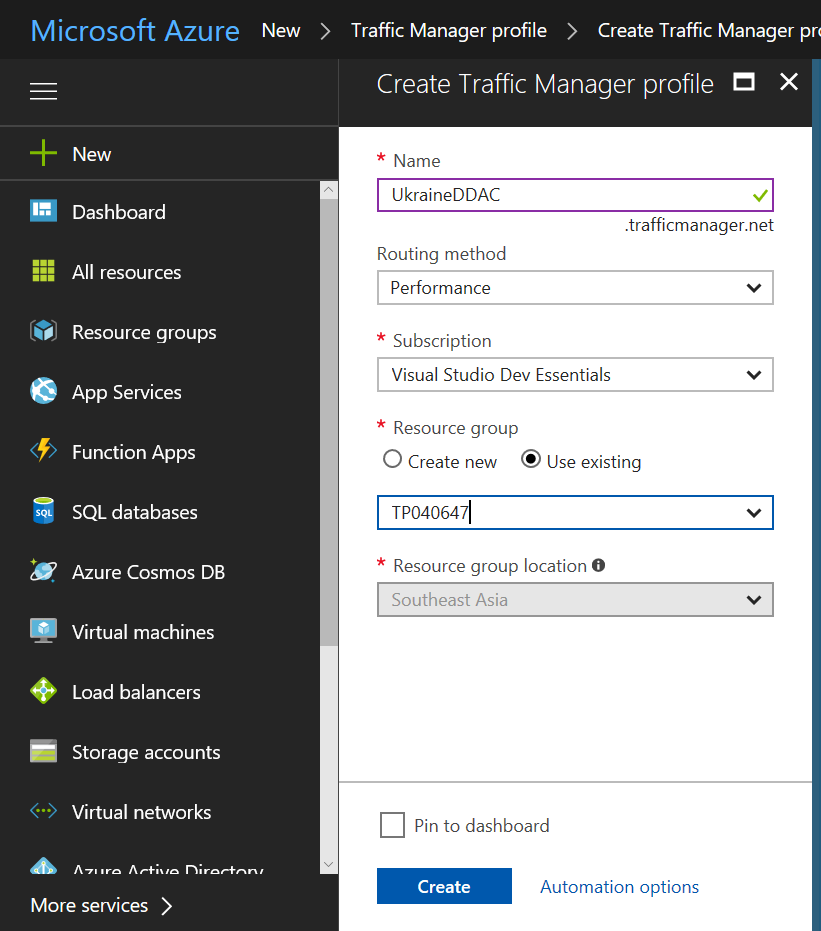


The web app requires a resource group and OS type. This is to let the developer to upload the web application through the Visual Studio and allow Azure to host the web application in cloud.

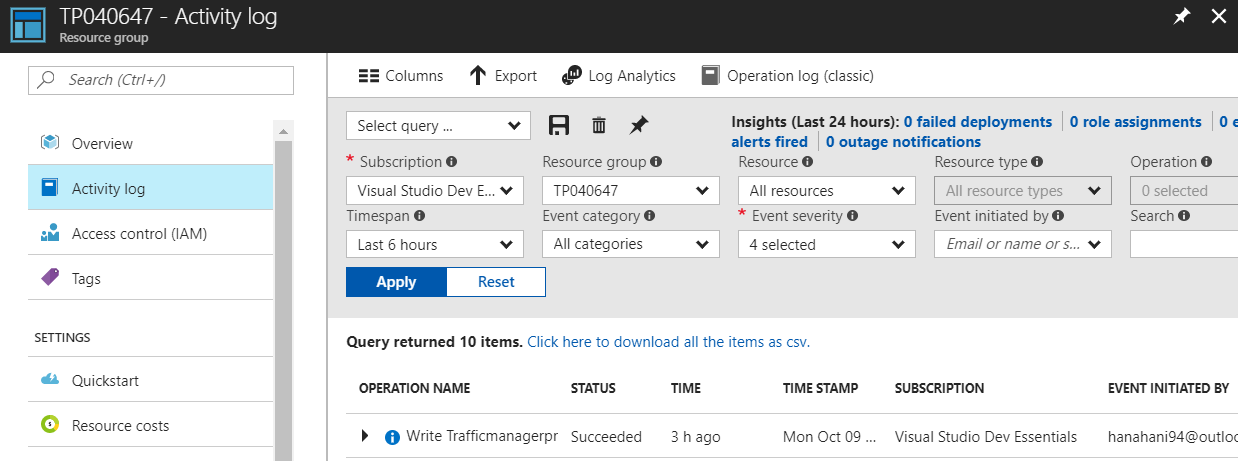
## Create Traffic Manger Profile



Developer will create a traffic manager profile to reduce the downtime in improve responsiveness of application that are deemed important. The traffic manager is to improve app availability with automatic failover, increase app responsiveness and seamlessly compile on premises and cloud

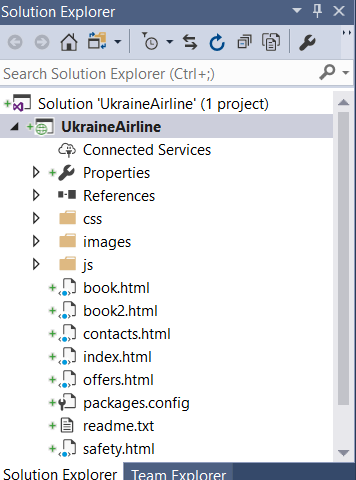


This is the details needed for create a traffic manger profile

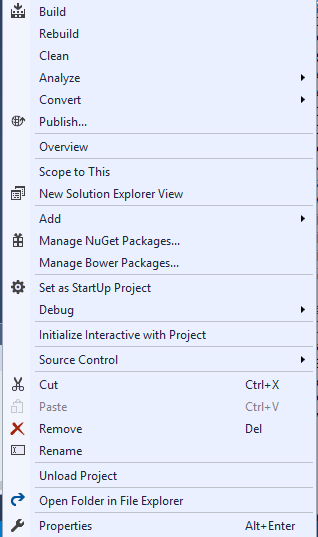


This activity log will display all the activity in the app service.

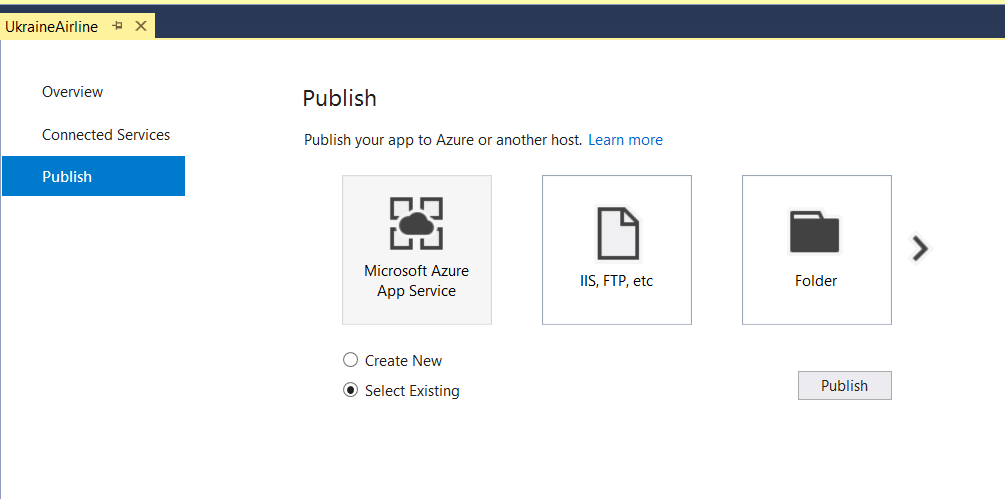
## Upload Web Application



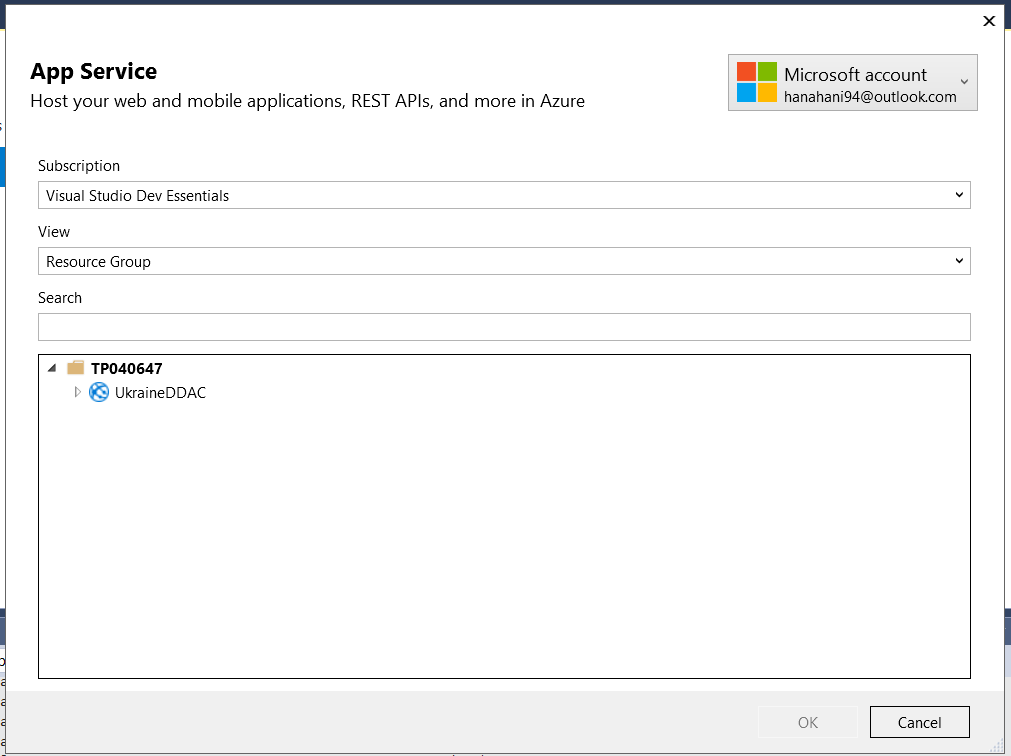
Developer has to go solution explorer to find your application in order to public the web application to Azure.



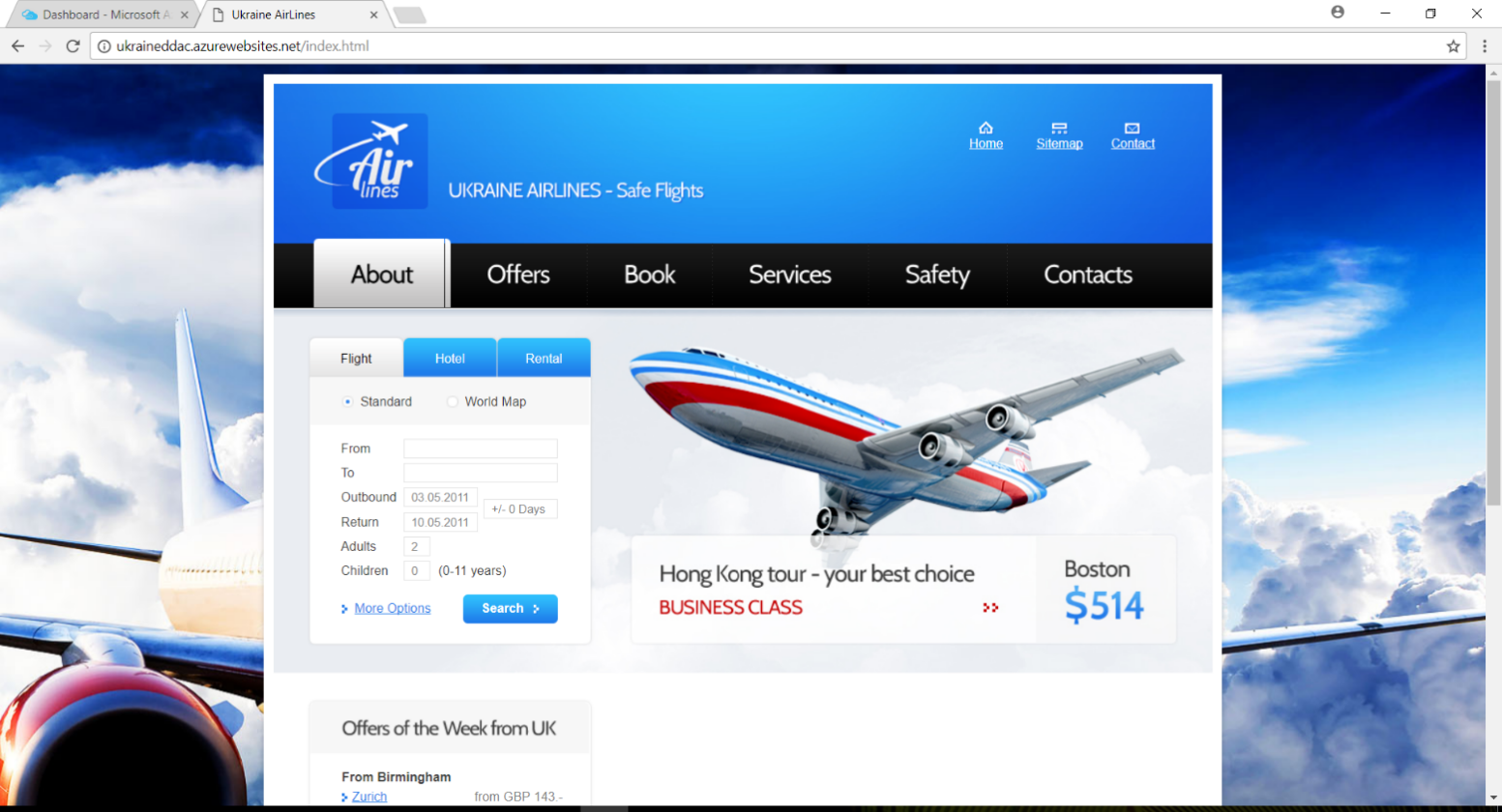
Right click the web application and find the publish field to public the website



After clicking the publish website, visual studio will display few option for public the website, for this situation, developer will select public the web application in Azure and select the resource group which create earlier.

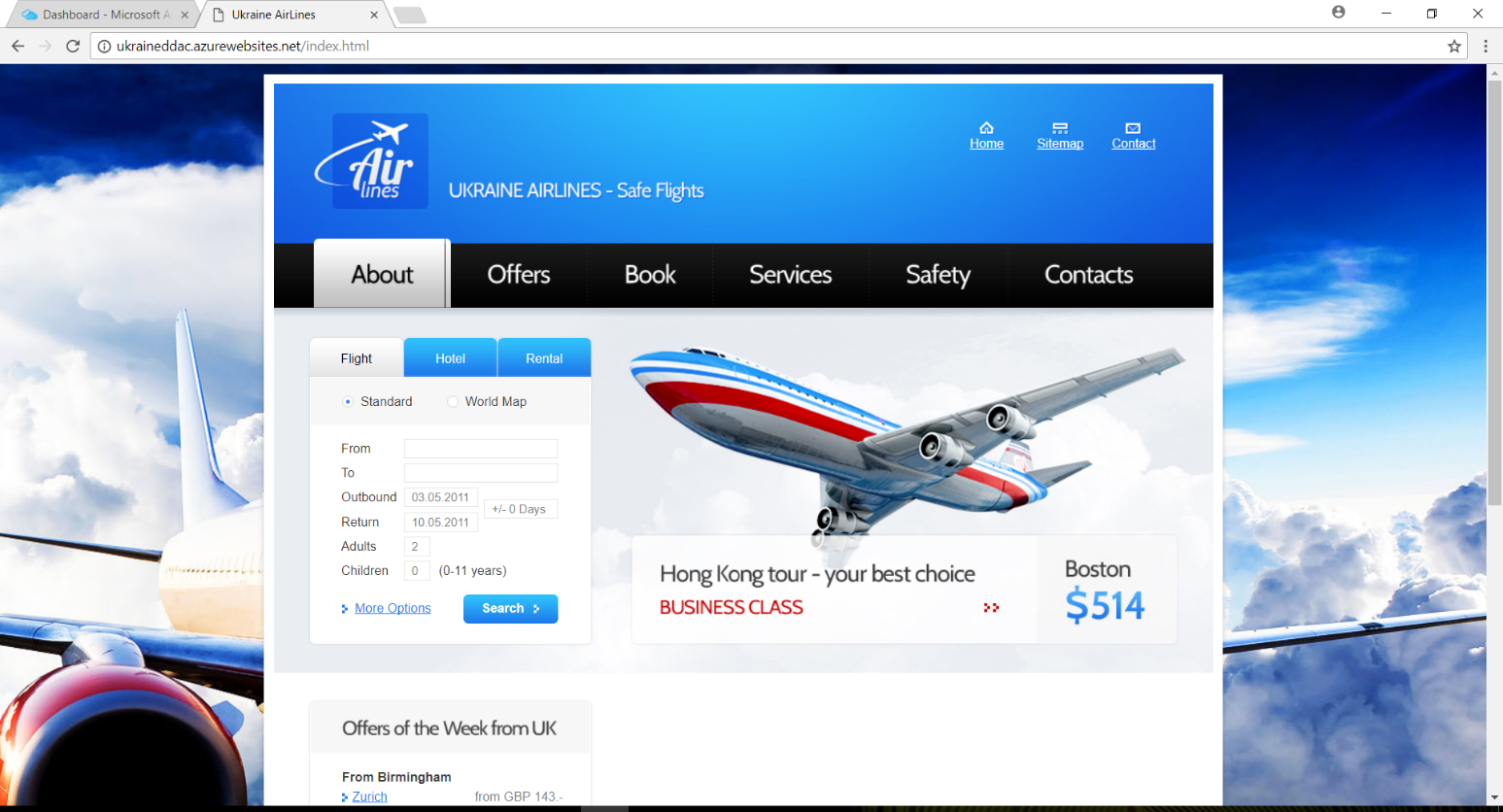


After that, developer has to select the right resource group and web application in order to public the web application to the Azure

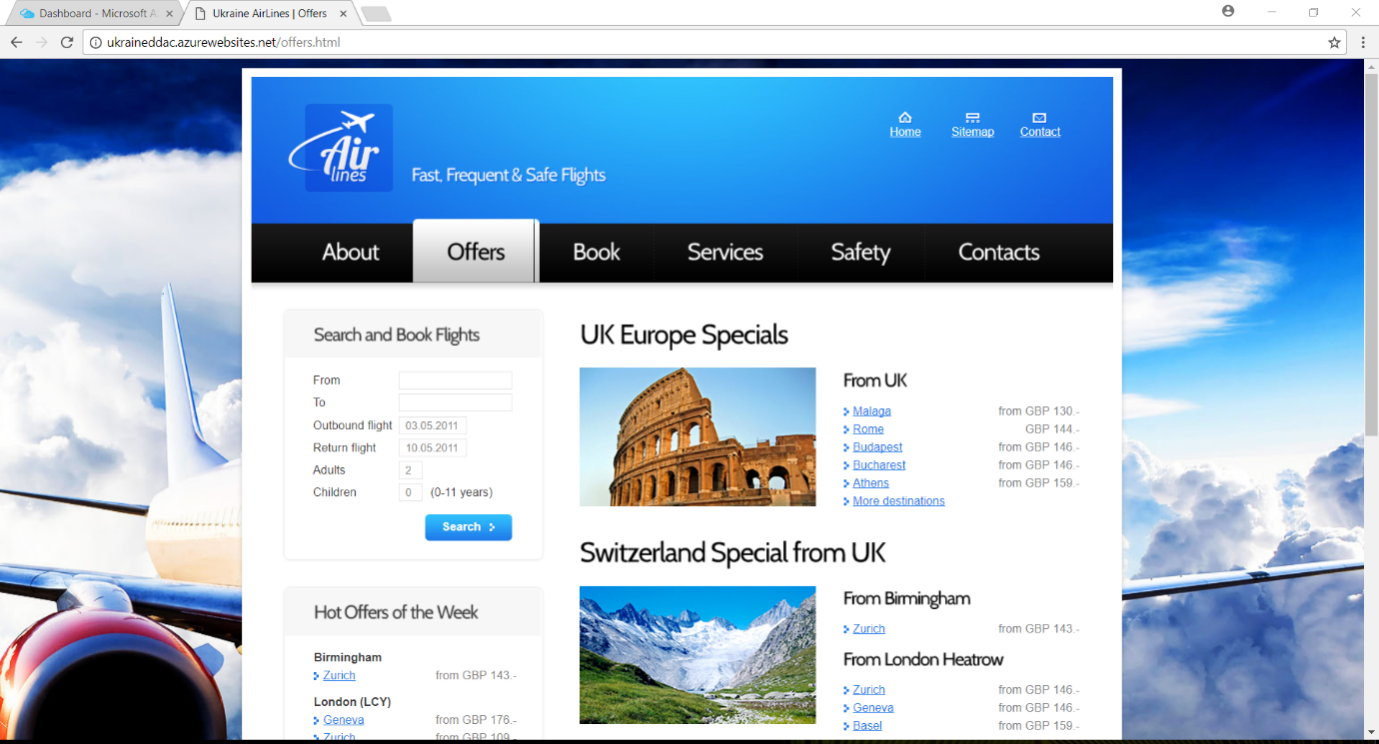


After successfully upload the website to Azure, it will pop out the home page of the website. The website link will show the azurenet.ne at the url.

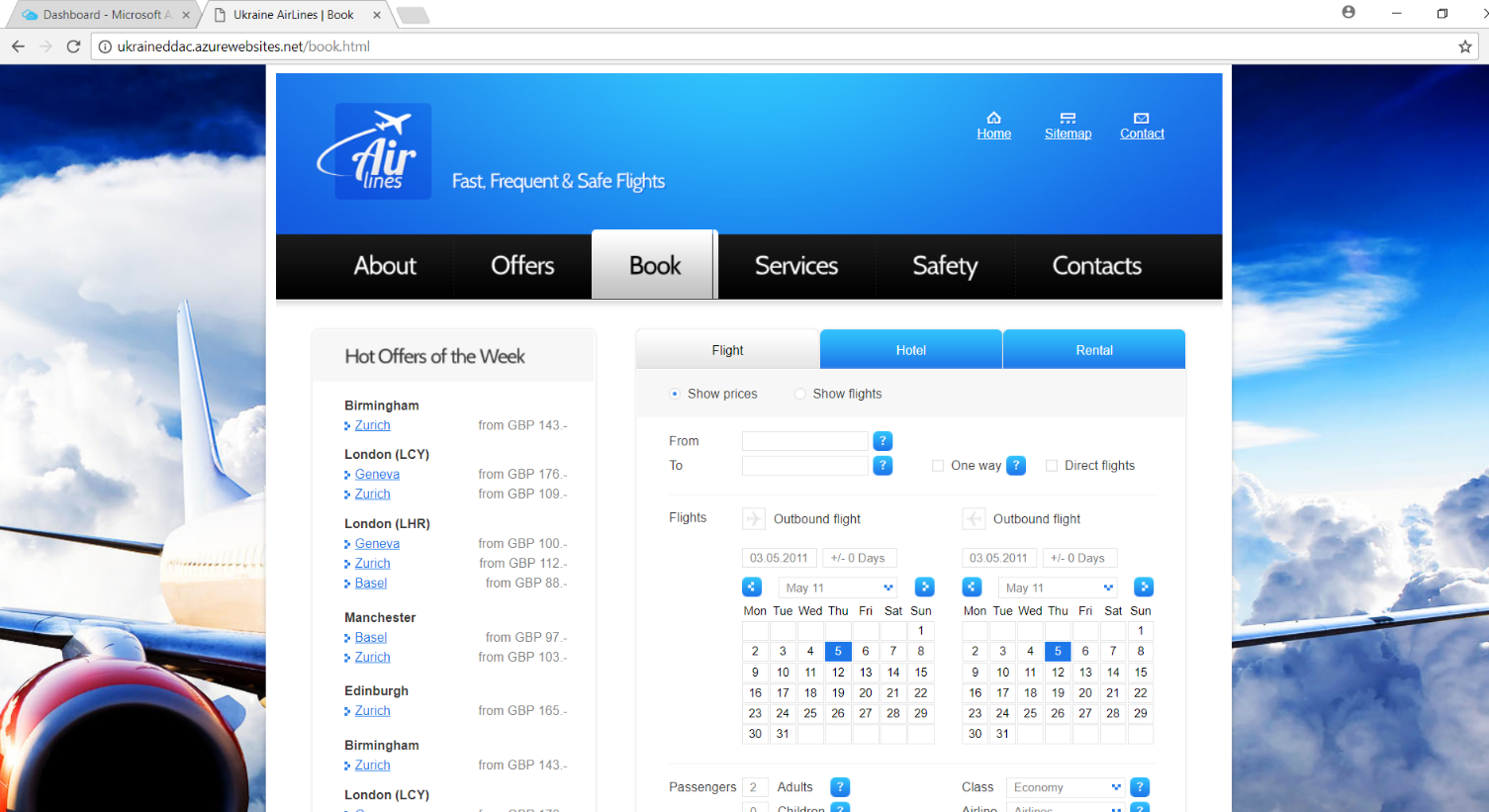
## Web application interface



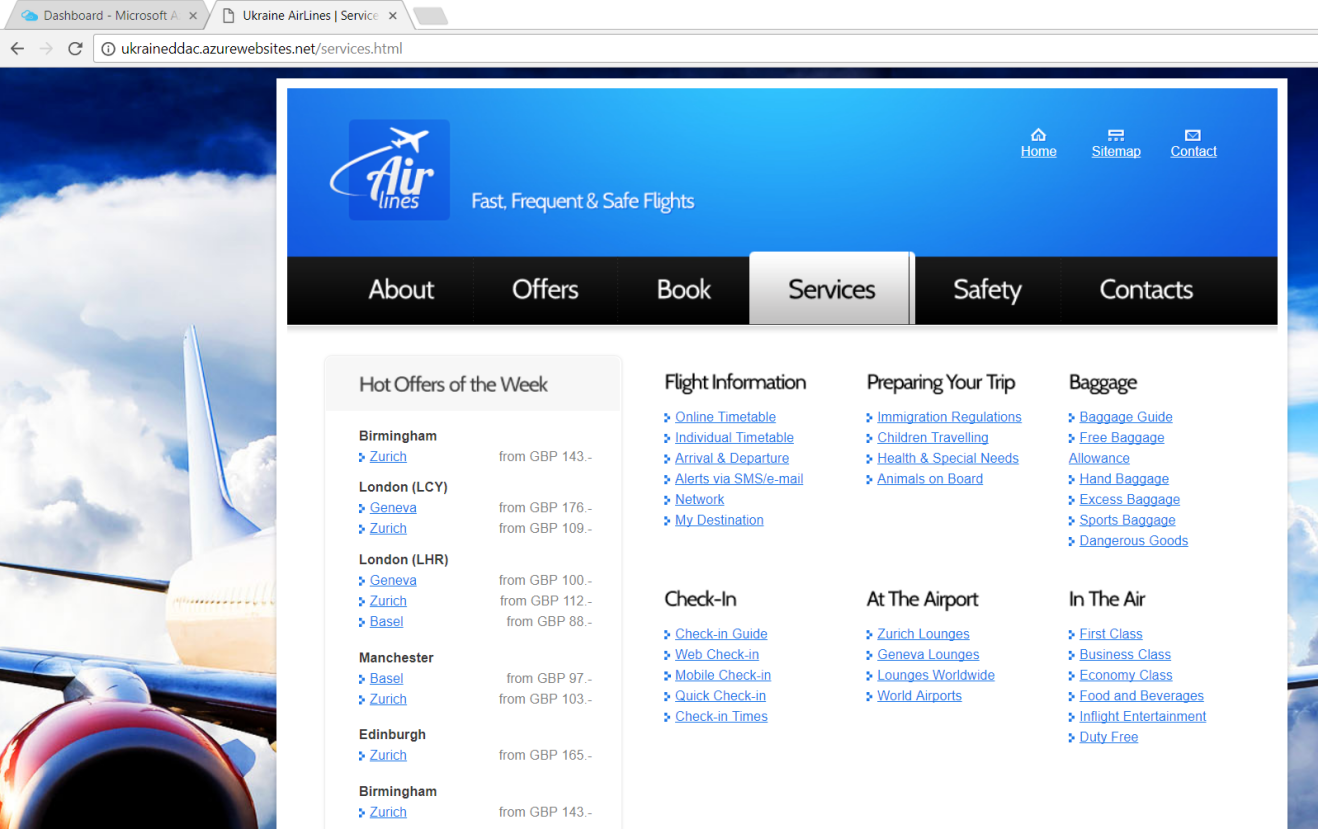
This is the web application home page. It will display the website logo of the company, the advertisement of the ticket price and searching flight.



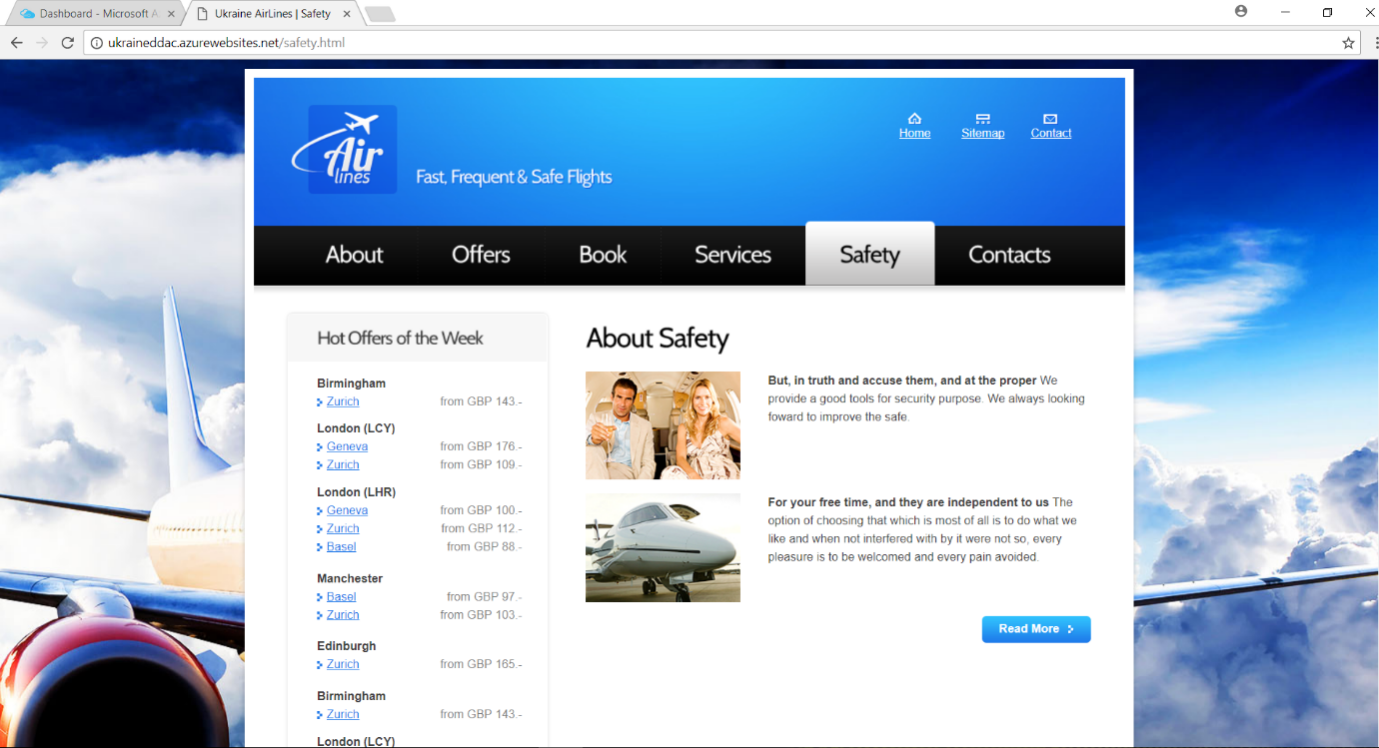
This is the offer page by Ukraine Airline. It will show the current offer of the ticket price.



This is the book flight ticket page. From here, passenger can choose their departure and arrival destination also the date. They also can choose how many people to book the flight ticket.



This page is about services that passenger that get the information of the Ukraine Airline services like Baggage Allowance.



This is the safety page that explain the safety of the Ukraine Airlines.

# Test plan & Testing Discussion

## Unit Testing

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case** | **Test Function** | **Expected Results** | **Actual Result** |
| **TC-001** | Login | Passenger able to login and access to their function | As expected |
| **TC-002** | Register | Passenger can register a new account | As expected |
| **TC-003** | Check availability | Passenger can check the status availability of seat | As expected |
| **TC-004** | Book ticket | Passenger can book flight ticket | As expected |
| **TC-005** | Choose seat | Passenger can choose their wish seat | As expected |
| **TC-006** | Payment | Passenger are able to make payment | As expected |
| **TC-007** | Cancel ticket | Passenger and admin are able to cancel the flight ticket that has been booked | As expected |
| **TC-008** | Check flight status | Passenger can check flight status | As expected |
| **TC-009** | Update flight schedule | Admin able to update flight schedule | As expected |
| **TC-010** | Validate card details | Bank able to validate passenger card details. | As expected |

## User Acceptance Testing

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Name of tester | |  | | | | | | | | | |
| Position | |  | | | | | | | | | |
| Time | |  | | | | | | | | | |
| Comment by tester | |  | | | | | | | | | |
|  | | | | | | | | | | | |
| Category | Poor | | | Medium | | | Good | | | Excellent | |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| Functionality |  |  |  |  |  |  |  |  |  |  |  |
| Interface Design |  |  |  |  |  |  |  |  |  |  |  |
| User Friendly |  |  |  |  |  |  |  |  |  |  |  |
| System Stability |  |  |  |  |  |  |  |  |  |  |  |
| System Flexibility |  |  |  |  |  |  |  |  |  |  |  |
| System Security |  |  |  |  |  |  |  |  |  |  |  |
| System Usability |  |  |  |  |  |  |  |  |  |  |  |

# Conclusion

As a conclusion, the developer able to improve the website that is prevented it from adequately serving customers beyond Ukraine in order to suit their situation. For this Ukraine Airline case, their need a user-friendly and a platform can fulfil the requirement based on the problem. After study the problem facing by Ukraine Airline, the developer decides to use the Visual Studio to develop a web solution. It is because Microsoft provides a user-friendly cloud platform and can fulfil the requirement to deploy the website; Visual Studio can easy deploy the website on the Microsoft Azure.

Besides, Microsoft Azure is the best choice to deploy the website because it has much service to use for the solution. Thus, developer can do version control in the Azure like rollback to the older version of the application when the latest have some bug and developer can monitor and security through Azure. Furthermore, after the system go to production, developer can have conducted testing, which in the end gave positive results as all the features were well validated. The developer can learn some new skill like programming skill and knowledge in Azure by deploying the system in the Microsoft Azure.

# References

Aryal, N. (2017). *How To Deploy Web App To Azure Using Visual Studio*. [online] C-sharpcorner.com. Available at: http://www.c-sharpcorner.com/article/how-to-deploy-web-app-to-azure-using-visual-studio/ [Accessed 8 Oct. 2017].

Azure (2017). *Azure Regions - Azure Speed Test*. [online] Azurespeed.com. Available at: http://www.azurespeed.com/Information/AzureRegions [Accessed 8 Oct. 2017].

G, T. (2009). *What's the difference between architecture and design? – Tom Graves / Tetradian*. [online] Weblog.tetradian.com. Available at: http://weblog.tetradian.com/2009/10/09/architecture-versus-design/ [Accessed 8 Oct. 2017].

Microsoft (2017). *Web App Service | Microsoft Azure*. [online] Azure.microsoft.com. Available at: https://azure.microsoft.com/en-us/services/app-service/web/ [Accessed 5 Oct. 2017].

Savill, J. (2014). *Understand Azure Traffic Manager*. [online] Windowsitpro.com. Available at: http://windowsitpro.com/windows-azure/understand-azure-traffic-manager [Accessed 7 Oct. 2017].

<https://github.com/hanahani94/Flight>

https://web.microsoftstream.com/video/5f104c96-a468-411b-9f53-9294a6d94460